基于运动目标特征的关键帧提取算法 *

田丽华，张　咪，李　晨†

(西安交通大学软件学院，西安 710049)

摘　要：针对运动类视频特征不易提取，且其关键帧结果中易产生较多漏检帧的问题，提出了基于运动目标特征的关键帧提取算法。该算法在强调运动目标特征的同时弱化背景特征，从而防止由于运动目标过小而背景占据视频画面主要内容所导致的漏检和冗余现象。首先根据视频帧熵值将颜色变化明显的帧作为部分关键帧，对颜色未发生突变的帧根据运动物体的尺度不变特征变换(SIFT)获得帧内运动目标的特征点。最后分别根据帧熵值及运动物体 SIFT 点分布提取视频关键帧。实验结果表明该算法所得关键帧结果集不仅漏检率较低且能够准确地表达原视频内容。

关键词：关键帧提取；混合高斯检测；SIFT 特征点；感知哈希

中图分类号：TP391.41　　doi: 10.3969/j.issn.1001-3695.2018.06.0363

Key frame extraction algorithm based on feature of moving target

Tian Lihua, Zhang Mi, Li Chen†

(*School of Software Engineering*, *Xi'an Jiaotong University*, *Xi'an* 710086, *China*)

Abstract: Motion features are difficult to extract which easily leads to missed and redundant frames in the result. In order to solve this problem, this paper proposes a method of key frames extraction based on feature of moving target. The method reduces redundant rate and missed rate of the result set by emphasizing the features of the moving target and weakening the background features of the frame. In this method, a frame with a burst entropy change is taken as part of the key frame firstly. Then, our method extracts the SIFT points of the moving target from the frame of which the entropy value has not suddenly changed. Finally, it extracts key frames according to the entropy and SIFT distribution respectively. Experimental results show that the miss rate of this algorithm is low. At the same time, key frame results can accurately and completely describe the main content of the video.

Key words: key frame extraction; gaussian mixed model detection; SIFT; perceptual hashing

0　引言

关键帧提取是视频检索的重要步骤，其提取结果质量好坏直接影响视频检索的效率。由于运动类视频中场景切换频繁，故其关键帧结果中存在较多漏检帧。运动类视频的关键帧提取方法多种多样。文献[1,2]提出通过分析光流场实现对运动场的分析从而提取关键帧。Ma 等人[3]提出在运动对象状态发生变化处提取关键帧[3]。Li 等人[4]提出了一种聚焦物体运动的关键帧提取算法，该方法将匀速运动的物体作为参照物，从而获得视频中运动目标并提取关键帧。针对视频中由于摄像机移动造成的物体运动， Guironnet 等人[5]提出通过检测摄像机运动提取关键帧，根据镜头运动方式排列顺序不同，灵活地调整关键帧提取方法[5]。以上方法对运动类视频的关键帧提取能力有限，当视频中存在快速运动的物体时，关键帧结果中易产生漏检帧。

由于局部特征能更好地保留帧语义，故目前存在较多根据局部特征提取关键帧的算法。其中 Hannane 等人[6]提出根据帧 SIFT 分布直方图提取关键帧。Barbieri 等人[7]提出根据固定大小窗口提取候选帧，再根据候选帧间 SIFT 特征距离提取关键帧，然而该方法所得候选帧无法准确地描述运动目标的变化过程。

结合运动类视频特点，本文提出了基于运动目标特征的关键帧提取算法。该算法首先根据帧熵值选取颜色发生突变处的帧作为部分关键帧。对熵值未发生突变的帧，根据混合高斯模型获得帧内运动物体并提取其 SIFT 特征，计算 SIFT 分布距离提取关键帧。根据运动物体 SIFT 分布特征提取关键帧，在强调帧内运动目标变化的同时弱化背景变化带来的影响，因此能够很好地捕获运动物体的变化过程。最后使用感知哈希算法获得帧指纹，计算其汉明距离去除结果中的冗余帧，以进一步提高关键帧对原视频的有效表达。

收稿日期：2018-06-22；**修回日期：**2018-08-07　　**基金项目：**国家自然科学基金项目（61403302）；西安交通大学科研业务基金项目（XJJ2016029）

作者简介：田丽华（1978-），女，陕西西安人，高级工程师，博士，主要研究方向为多媒体技术；张咪（1992-），女，陕西西安人，硕士研究生，主要研究方向为关键帧提取；李晨（1981-），女（通信作者），陕西西安人，讲师，博士，主要研究方向为多媒体技术（lhtian@xjtu.edu.cn）.

1　理论基础

1.1　混合高斯模型

混合高斯建模(GMM)适用于对背景复杂的视频进行建模，从而检测出视频中的运动目标[8,9]。根据混合高斯模型检测运动物体，首先计算出帧中像素点在某时间段内样本值的概率密度等信息，接着根据统计差分中常见原则判定各像素点是否为帧背景内容。每读取新的视频帧时，首先根据当前帧更新模型，再遍历待处理帧中的各像素点，判断其是否为背景点。完成帧各像素点的处理后即能够得到该帧背景模型，然后根据背景得到帧前景。混合高斯模型能够很好地对内容复杂的视频进行背景建模。

1.2　SIFT 特征

SIFT(scale invariant feature transform)特征点是一种稳定的局部特征。选取物体表面具有代表性的点来描述物体，这些点通常称为兴趣点[10]，如较暗区域亮度较高的点，明亮区域中亮度较低的点，物体边缘的特殊点等都可能成为兴趣点。兴趣点不会因为图像缩放或者旋转而产生较大变化，同时对于光线改变以及噪声干扰都具有一定鲁棒性。

1.3　感知哈希

图像感知哈希技术也称为数字指纹，是对多媒体信息的总结。感知哈希是一类从多媒体表示到哈希值的单向映射，各图像都有其对应的指纹。

感知哈希算法首先去掉图片细节，忽略图像尺寸和比例差异，仅保留结构、阴影等基本信息。接着计算各帧灰度图的离散余弦变换（DCT）系数矩阵，仅保留左上角的 8*8 矩阵。计算所有像素灰度均值，比较矩阵各像素点与均值大小，大于或等于均值记为“1”，否则记为“0”，最后生成帧指纹。根据帧指纹计算汉明距离，判断帧相似性。

2　基于运动目标特征的关键帧提取

为强调视频中运动目标的变化，本文提出根据运动目标特征提取关键帧的算法。该算法首先计算帧熵值，选择颜色突变的帧作为部分关键帧。对颜色未发生突变的帧，根据帧内运动物体的 SIFT 分布提取关键帧。最后根据感知哈希算法去除结果集中的冗余帧。算法核心步骤流程如图 1 所示。

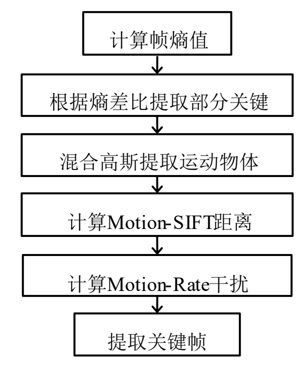


图 1　关键帧提取过程

计算帧熵值：读入视频帧序列，根据式(1)(2)所示计算帧熵值，将熵值突变处的帧选为部分关键帧。

$$p(x_i) = Num(x_i) / m*n \tag{1}$$

$$H = -\sum_{i=0}^{255} p(x_i)*\log(p(x_i)) \tag{2}$$

其中：x_i 代表视频帧像素点 i 处的灰度值，m 和 n 分别为帧的高和宽。$Num(x_i)$代表帧内灰度值为 x_i 的像素点总数量，$p(x_i)$为 x_i 在视频帧中出现的概率。

a)对熵值未发生突变的帧，根据混合高斯模型（GMM）检测出帧内运动物体并对其进行腐蚀膨胀处理。

b)将帧分成多个扇形区，计算相邻帧 SIFT 分布距离（Motion-SIFT）。

c)对距离突变的帧，计算运动目标在帧内所占像素百分比(Motion-Rate)，判断当前帧是否受到背景运动干扰。

d)使用感知哈希算法去除结果集中的冗余帧。

2.1　混合高斯提取运动目标特征

读入帧序列计算相邻帧熵值比，取熵突变处的帧为部分关键帧。对熵值未突变的帧，进一步计算帧内运动物体的变化程度判断其是否为关键帧，具体处理过程如下：

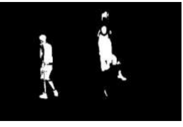
a)对熵值未发生突变的帧，根据 GMM 获得视频帧内的前景运动物体，结果如图 2(b)所示。

b)为使获得的帧内运动目标轮廓更加清晰，对帧前景内容进行腐蚀膨胀处理。如图 2(c)为对图 2(b)中的帧进行腐蚀膨胀处理的结果。

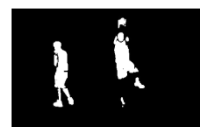
c)逐个获得视频中所有熵值未发生突变帧的运动内容，最后得到一组仅包含运动物体的二值化图像。



(a)原视频帧



(b)帧内运动物体



(c)形态处理图像

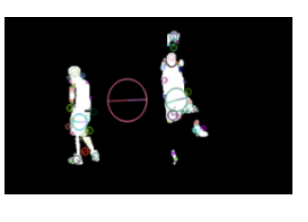
图 2　帧内运动物体

提取视频中各帧的前景内容，获得仅包含运动物体的二值化图像，其中白色部分表示视频中的运动物体，黑色区域为帧的背景内容。接着提取帧运动内容的 SIFT 特征，根据相邻帧 SIFT 点分布计算帧间距离。

如图 3(a)所示为根据混合高斯检测到的视频帧内的运动物体完成腐蚀膨胀处理后的结果。提取并标记出图 3(a)的 SIFT 点结果如图 3(b)所示。由图 3(b)中特征点分布可得，SIFT 点能够有效标记出帧内运动物体的位置。



(a)运动目标



(b)SIFT 特征点

图 3　运动物体 SIFT 特征

2.2 Motion-SIFT 距离计算

计算出帧内运动物体的 SIFT 特征点(Motion-SIFT)，根据其在各扇形区的分布情况提取关键帧。计算 SIFT 分布距离首先需获得帧内运动物体的 SIFT 特征点，然后把各帧划分成多个扇形区域。接着将 SIFT 特征点转换到极坐标系下，统计各扇形区的 SIFT 点个数并计算帧间距离提取关键帧。Motion-SIFT 距离计算过程如下：

1）帧扇形分区

为统计视频中运动物体的 SIFT 分布情况，本文算法将视频帧分成多个扇形区。由于运动类视频的主要内容可能出现在帧的任何区域，故根据不同的半径和角度将视频帧分成多个扇形区。

h 为待分区视频帧的高，*w* 为帧宽。直角坐标系下视频帧中心为 O(x_o,y_o)，其横坐标 $x_o=w/2$，纵坐标 $x_o=h/2$。以视频帧中心点为圆点，根据不同半径和角度将帧划分成多个扇形区，结果如图 4 所示。

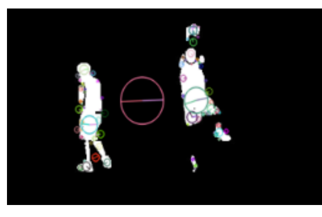
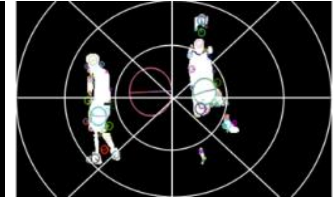



图 4　帧扇形分区

视频帧扇形分区具体步骤如下：

a）首先获得视频帧的高 h 和宽 w，以视频帧中心 O 为圆心，分别以 $r_1=w/6$，$r_2=w/3$，$r_3=w/2$ 为半径作圆，视频帧被三个同心圆分为了 3 个环形区域。

b）根据不同角度对 SIFT 特征点划分：以 45 度为单位，使用四条过视频帧中心 O 点的直线，将帧根据不同的角度划分成 8 个范围。最终根据角度和距离不同，视频帧被分成 24 个扇形区。

2）坐标系变换

根据上述步骤完成帧分区后，计算帧中各区 SIFT 特征点的分布数量。统计各区特征点个数并将所有特征点从直角坐标系转换为以原视频帧中心 O 点为极心的极坐标下。(x_i,y_i)表示第 i 个 SIFT 特征点的直角坐标，该特征点(x_i,y_i)的极坐标表示形式为(r_i,θ_i)。

如式(3)所示,首先计算出坐标为(x_i,y_i)的特征点以帧中心为原点的直角坐标系下的坐标值(x_i’,y_i’)。

$$\begin{cases} x_i' = x_i - w/2 \\ y_i' = y_i - h/2 \end{cases} \tag{3}$$

接着将各特征点转换为以视频帧中心点 O (w/2,h/2)为极心的极坐标系下，分别根据式（4）(5)计算出极坐标系下 SIFT 点(x_i,y_i)的对应的极坐标(r_i,θ_i)。

$$\theta_i = \arctan\left(\frac{(y_i' - y_o')}{(x_i' - x_o')}\right) \tag{4}$$

$$r_i = \sqrt{(x_i' - x_o')^2 + (y_i' - y_o')^2} \tag{5}$$

重复上述步骤获得所有特征点的极坐标，统计特征点在各扇形区内的分布数量，计算帧间距离提取关键帧。

3）Motion-SIFT 分布统计

计算 SIFT 分布距离，首先需统计各区特征点个数并计入 Count[i][j]。其中 i,j 分别代表 SIFT 特征点半径 r 范围，和角度 θ 范围。当 r=0 时表示该像素点处于最小圆(r_1=w/6)内，当 r=2 时表示该像素点在最大的圆(r_3=w/2)内或该圆之外，当 0<θ<π/4 时，与该特征点角度对应的数组 j 值为 0。

获得帧内各扇形区 SIFT 点个数后，接着根据特征点 S_i 极坐标(r,θ)确定其所在区域。由式(6) (7)可计算出特征点(r,θ)所属的扇形区第(i,j)区，将此特征点计入 Count[i][j]。

$$i = \frac{r}{(w/6)} \tag{6}$$

$$j = \begin{cases} \dfrac{\theta}{(\pi/4)} & x'>0 \ y'>0 \\ \dfrac{\theta+\pi}{(\pi/4)} & x'<0 \\ \dfrac{\theta+2*\pi}{(\pi/4)} & x'>0 \ y'<0 \end{cases} \tag{7}$$

逐个判断运动物体 SIFT 特征点所在区域，得到统计扇形特征点个数的二维数组，该数组描述运动目标的特征点分布情况。

4）计算帧间距离

根据运动物体提取 Motion-SIFT 特征点，从而计算其帧间距离。设视频中的连续两帧分别为 f_k，f_{k+1}，其对应的 Motion-SIFT 特征点分布数组分别为 $Count_k$[i][j]和 $Count_{k+1}$[i][j]。如式(8)所示，计算两视频帧特征点分布距离（SiftCountDiff）。

$$SiftCountDiff(f_k, f_{k+1}) = \sqrt{\sum_{i=0}^{2}\sum_{j=0}^{7}((Count_k[i,j] - Count_{k+1}[i,j])^2)} \tag{8}$$

由于运动类视频中可能将背景误检为运动目标，从而造成 SiftCountDiff 差值骤增。采用平均值作为阈值提取关键帧容易造成运动变化较快处的关键帧漏检。因此本文算法使用相邻帧间特征点分布距离的比值来衡量运动特征的变化程度，其计算过程如式(9)所示。

$$MotionChange(k, k+1) = SiftCountDiff(k+1) / SiftCountDiff(k) \tag{9}$$

视频中相邻两帧的 Motion-SIFT 分布距离比值记为 MotionChange，根据相邻帧比值衡量帧特征距离，判断是否为关键帧。当 MotionChange(f_k, f_{k+1})值发生突变时，进一步判断当前帧 f_{k+1} 是否受到背景运动干扰。若未受背景干扰则将该帧选为关键帧加入关键帧集合中；否则依次比较其后两相邻帧的特征点分布距离，直到遍历完所有帧。根据 Motion-SIFT 距离提取关键帧伪代码如下：

方法：帧熵值及 Motion-SIFT 分布距离提取关键帧

```
1:  Count[i][j]←0     // number of feature points of each sector
2:  for  n =0→iNum do       // number of frames of video
3:      extract Motion-Object of frame by GMM
```

```
    extract SIFT points of frame
    divide frame into sectors
    for  k =0→SiftNum  do
        getting r and θ
        calculating i and j, Count[i][j]++
    end for
    call formula(4-9) calculate SiftCountDiff
    call formula(4-10) calculate MotionChange
    if MotionChange>α  then
     if  MotionRate>β  then
            goto(2)
        else
            add this frame into keyframe set
        end if
    end if
end for
```

2.3 Motion-Rate 计算

通过混合高斯检测出的视频中的运动物体，其中白色部分为帧内运动目标，黑色为帧背景内容。计算白色像素点在整个帧中的占比(Motion-Rate)，根据该占比判断运动内容的变化情况。当相邻帧间的 Motion-Rate 值在局部范围发生较大变化时，则判定为有大量背景信息介入。

判断关键帧提取过程中是否受到运动背景干扰，首先需统计出帧内所有白色像素点的个数记为 MotionCount。然后计算当前待判断帧与上一个关键帧中运动目标的像素总数比值(MotionChange)。相邻关键帧间运动目标像素总数比值如式(10)所示。

$$MotionChange(i) = MotionCount(i) / MotionCount(cur) \tag{10}$$

运动类视频中，帧内运动物体的变化占整帧内容百分比较小。故本文根据混合高斯检测出的帧中运动目标在视频帧中所占百分比(Motion-Rate)以及待判断关键帧像素总数的变化幅度来判断是否存在背景干扰。运动目标在帧内所占百分比计算如式(11)所示。

$$MotionRate = MotionCount / Allpixel \tag{11}$$

根据帧运动目标像素占比以及该帧与其前一帧的变化程度即可判断当前帧是否受到运动背景干扰。当存在背景干扰时继续判断下一候选帧是否为关键帧，未受到背景干扰时则将该帧加入关键帧集合。

最后，使用感知哈希算法去除所得关键帧集合中的冗余帧。计算结果集中相邻帧间的汉明距离，当两帧距离小于 n 时则认为当前存在一帧冗余。

3 实验结果与分析

为检测本文算法的有效性，本文在 Windows 7 操作系统下英特尔 Core i3，2GB 内存的计算机上进行，选择 Visual Studio 2010 及 OpenCV3.0 作为开发平台实现并测试该算法。实验中选取了多个运动视频进行测试，其中包括足球比赛、篮球比赛、运动教学、体操、滑冰以及击剑等类型视频。

3.1 评价标准

查准率（P）和查全率（R）是对结果集的有效评价标准。其中查全率用于衡量关键帧的漏检情况如式(12)所示。查准率反映提取结果的准确性如式(13)所示。

$$R = \frac{N_c}{N_c + N_m} \times 100\% \tag{12}$$

$$P = \frac{N_c}{N_c + N_f} \times 100\% \tag{13}$$

其中:N_c，N_m，N_f分别表示结果中正确提取的关键帧数量，漏检帧个数和误检个数。由于关键帧提取过程中 R 值和 P 值难以兼顾，因此为综合度量结果集的有效性，本文使用两者的调和平均值 F 评价结果集，其定义如式(14)所示。

$$F = \frac{2 \times R \times P}{R + P} \tag{14}$$

3.2 实验结果

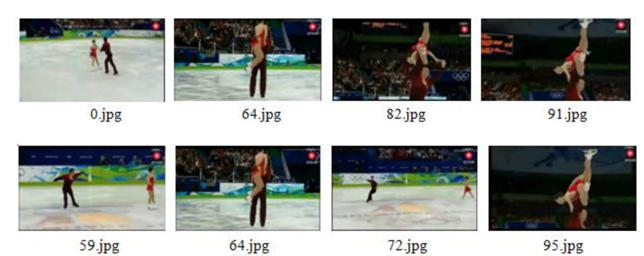
3.2.1 关键帧提取视觉结果

实验中本文截取了 15 个运动类视频片段进行测试，并确定各参数取值，对关键帧提取结果进行分析和总结。实验结果显示当汉明距离 n 为 5 时，对相似帧判断最准确；MotionChange 比值大于 3 时，能够判定该帧相对于前一帧发生了突变。

为验证本文算法的有效性，将本文算法与基于 SIFT 分布直方图的关键帧提取算法（SIFT-HD）[6]进行比较，其视觉结果如图 5~7 所示。



(a) 基于帧 SIFT 分布直方图关键帧提取结果



(b)基于运动目标 SIFT 特征关键帧提取结果

图 5 花样滑冰视频的关键帧结果集

图 5(a)中检测到 4 个关键帧，图 5(b)中检测到 8 个关键帧，其中图 5(a)和图 5(b)都存在 1 帧冗余。根据原视频，运动员完成了多组动作，包括两次分开和三次次共同完成的动作。图 5(a)仅反映出两次共同完成的托举动作，故图 5(b)对视频内容表达更加准确。

如图 6 所示 SIFT-HD 算法检测到 2 个关键帧，本文算法检测到 8 个关键帧帧。原视频中，右方选手攻击两次防守一次，左方选手攻击一次防守两次。如图 6(b)所示，本文结果能完整的表达两运动员攻守情况,而图 6(a)无法准确地表达原视频内容。



(a)基于帧 SIFT 分布直方图关键帧提取结果



(b) 基于运动目标 SIFT 特征关键帧提取结果

图 6　击剑关键帧结果

如图 7(a)所示，该关键帧结果仅能描述出运动员三步上篮过程，丢失了该运动员尝试投篮被拦下的动作。图 7(b)中则可完整描述整个过程。



(a) 基于帧 SIFT 分布直方图关键帧提取结果



(b) 基于运动目标 SIFT 特征关键帧提取结果

图 7　篮球教学片断关键帧结果

由图 5~7 可得本文算法相较于文献[6]算法对原视频内容的表达能力更强，对原视频内容的表达更加准确。

3.2.2 实验结果分析和对比

将本文算法分别与基于帧 SIFT 分布直方图（SIFT-HD）[6]及基于帧 SIFT 特征(KS-SIFT)[7]的关键帧提取算法进行对比。计算结果的冗余率，漏检率及其调和值。由于运动视频难以人工标定关键帧，故将本文算法和对比算法的结果互为参考标准判断各自漏检帧。本文算法与 SIFT-HD 算法[6]关键帧结果的漏检及冗余帧统计结果如表 1 所示。

表 1　本文与 SIFT-HD 算法漏检冗余对比

视频	总帧	结果帧总数		漏检帧		冗余	
		文献[6]	本文算法	文献[6]	本文算法	文献[6]	本文算法
体操	155	8	9	3	1	2	1
篮球 1	149	14	12	0	0	2	0
篮球 2	115	4	8	3	0	1	1
篮球 3	165	7	13	4	0	1	3
足球 1	85	4	5	1	0	0	0
足球 2	82	19	16	1	0	7	3
足球 3	240	5	5	1	0	1	0
足球 4	102	8	10	2	0	1	1
击剑	125	2	10	6	0	0	2

由表 1 可知本文算法和 SIFT-HD 算法所得的结果中都存在少量冗余，但本文算法的漏检帧少于文献[6]。SIFT-HD 算法检测到的关键帧都包含在本文算法结果中，然而本文算法获得的关键帧结果中存在部分关键帧 SIFT-HD 算法未能检测到。故本文算法所得结果对原视频的重现程度更好。

将本文算法与基于帧 SIFT 特征的关键帧提取算法(KS-SIFT)[7]的关键帧结果进行对比。KS-SIFT 算法和本文算法所得结果的漏检帧及冗余帧的统计情况如表 2 所示。

表 2　与 KS-SIFT 算法漏检冗余对比

视频	总帧	结果帧		漏检帧		冗余帧	
		文献[7]	本文算法	文献[7]	本文算法	文献[7]	本文算法
体操	155	7	9	3	1	1	1
篮球 1	149	8	12	4	0	0	0
篮球 2	115	5	8	2	0	0	1
足球 1	85	4	5	1	0	0	0
足球 2	82	4	16	9	0	0	3
足球 3	240	4	5	1	0	0	0
足球 4	102	6	10	3	0	0	1
击剑	125	5	10	3	0	0	2

由表 2 可知本文算法和 KS-SIFT 算法[7]所得结果中的冗余帧都很少。本文算法的结果中漏检帧极少，而参考文献算法存在较多漏检帧。因此在冗余帧都较少的情况下，本文算法的查全率更高对视频内容的表达更加完整。

对比本文算法与 KS-SIFT 算法以及 SIFT-HD 算法，计算三种算法关键帧结果的 F 值，其统计结果如图 8 所示。

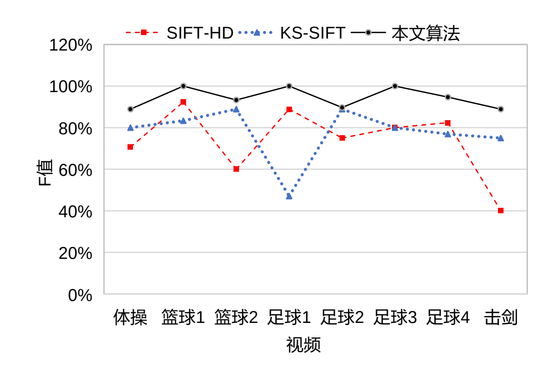


图 8　三种算法结果集 F 值对比

由图 8 可知本文算法的综合表现更优，相对于其他两种算法，本文算法所得结果的 F 值更加稳定。其中 SIFT-HD 算法和 KS-SIFT 算法的 F 值总体情况不相上下，都存在个别视频的 F 值低于 50%。基于帧 SIFT-HD 关键帧提取算法[6]和 KS-SIFT 算法[7]提取出的关键帧结果集中都存在较多漏检的关键帧，故其 F 值略低于本文算法。

综上本文算法提取出的关键帧能完整的表达出原视频的主要内容，而基于帧 SIFT 分布直方图关键帧提取算法所得关键帧集合对视频中部分内容的表达不够完整，会造成部分视频内容丢失。

本文进一步比较了三种算法的运行效率，实验中对上述 8

段运动类视频进行测试，帧大小均为 480*272。三种算法各自的帧处理平均时间如表 3 所示。

表 3 算法处理时间比较

算法	帧平均时间/s
KS-SIFT	0.018
SIFT-HD	0.026
本文算法	0.035

由表 3 可知 KS-SIFT 算法耗时最少，其原因在于 KS-SIFT 算法选取固定位置的帧为候选帧，再从候选帧中选取关键帧，能够极大地降低所需处理的数据量。然而此方法虽简单快捷，但不够灵活，候选帧的选取往往存在偏差。SIFT-HD 算法是根据帧 SIFT 分布直方图提取关键帧，较 KS-SIFT 算法肯定更为耗时，但效果一般好于 KS-SIFT 算法。相较于 SIFT-HD 算法，本文算法首先根据混合高斯建模获得帧内运动物体，然后提取运动物体的 SIFT 特征点。该方法减少了所需处理的 SIFT 特征点个数，但是在高斯混合提取运动目标时需要一定耗时，故其执行时间略高于 SIFT-HD 算法。虽然本文算法耗时略高，但其所得的关键帧结合对原视频的表达更加准确，且耗时在相同数量级，满足工程要求。

4 结束语

针对运动类视频，本文提出了一种基于运动目标特征的关键帧提取算法。该算法强调视频中运动物体的变化，弱化视频中运动背景的改变，从而防止由于背景内容过多或过于复杂而造成运动目标特征不明显导致的关键帧漏检。实验结果表明，即使在运动背景较为复杂运动目标占据帧百分比较小时，该算法依然能够较好地检测出运动目标的变化。基于运动目标特征的关键帧算法所得的关键帧结果，对于原始视频内容还原度高，能够完整准确地表达原视频内容。

参考文献：

[1] 崔智高，王华，李艾华，等. 动态背景下基于光流场分析的运动目标检测算法 [J]. 物理学报, 2017, 66 (8): 97-104. (Cui Zhigao, Wang Hua, Li Aihua, *et al.* Moving object detection based on optical flow field analysis in dynamic scenes [J]. Acta Physica Sinica, 2017, 66 (8): 97-104.)

[2] Shao Ling, Ji Ling. Motion histogram analysis based key frame extraction for human action//activity representation [C]// Proc of Canadian Conference on Computer and Robot Vision. Washington DC: IEEE Computer Society, 2009: 88-92.

[3] Ma Yanzhuo, Chang Yilin, Yuan Hui. Key-frame extraction based on motion acceleration [J]. Optical Engineering, 2008, 47 (9): 957-966.

[4] Li Congcong, Wu Yita, Yu S S, *et al.* Motion-focusing key frame extraction and video summarization for lane surveillance system [C]// Proc of IEEE International Conference on Image Processing. 2010: 4329-4332.

[5] Guironnet M, Pellerin D, Guyader N, *et al.* Video summarization based on camera motion and a subjective evaluation method [J]. Eurasip Journal on Image & Video Processing, 2007, 2007 (1): 060245.

[6] Hannane R, Elboushaki A, Afdel K, *et al.* An efficient method for video shot boundary detection and keyframe extraction using SIFT-point distribution histogram [J]. International Journal of Multimedia Information Retrieval, 2016, 5 (2): 89-104.

[7] Barbieri T, Goularte R. KS-SIFT: a keyframe extraction method based on local features [C]// Proc of IEEE International Symposium on Multimedia. 2015: 13-17.

[8] 黄东军，杨颖华. 基于改进混合高斯模型的运动物体检测研究 [J]. 计算机应用研究, 2017, 34 (6): 1862-1866. (Huang Dongjun, Yang Yinghua. Modified Gaussian mixture background model for moving object detection [J]. Application Research of Computers, 2017, 34 (6): 1862-1866.)

[9] 於正强，潘赟，宦若虹. 一种结合帧差法和混合高斯的运动检测算法 [J]. 计算机应用与软件, 2015 (4): 129-132. (Yu Zhengqiang, Pan Yun, Huan Ruohong. A moving detection method combining frame diffrence and gaussian mixture model [J]. Computer Applications and Software, 2015 (4): 129-132.)

[10] 李海洋，文永革，何红洲. 一种改进的 SIFT 特征点检测方法 [J]. 计算机应用与软件, 2013 (9): 147-150. (Li Haiyang, Wen Yongge, He Hongzhou. An improved SIFT feature point detection method [J]. Computer Applications and Software, 2013 (9): 147-150.)